Redazione: Rossella Torretta
Fotografie di Mimmo Capurso
progetto grafico di G.C. Greguoli

sommario

per cominciare ...

TUTTO BENE ...?

Ciao. Ciao a tutti i nostri vecchi ed affezionati lettori ma, soprattutto, un saluto di benvenuto ai nostri nuovi amici dell'MSX che, presto, ci auguriamo, formeranno una folta schiera. Finalmente, direte voi, in edicola con il primo numero!
16/MSX: *la nuova rivista in questione.*
Voi, amici del C-16 che ci avete già seguito fin qui su L.M. Vic 16/20, sapete già chi siamo e come si articola la nostra rivista.
Spendiamo allora qualche parola in più per chi invece possiede il computer MSX e non ci conosce affatto.
Cosa vi posso dire? Che questa è una rivista tutta speciale, piena zeppa di novità, di rubriche, di cose interessanti e... beh, vedrete da soli!
Ogni mese troverete nella vostra edicola giochi e programmi di utilità per il vostro computer, un meraviglioso corso a puntate per imparare a programmare in linguaggio macchina ed arrivare a capire tutti i segreti più nascosti del vostro computer.
Troverete inoltre una pagina di recensioni dedicata alle ultime novità in campo "computeristico", una pagina di "mercatino" dove potrete comunicare con altri amici per vendere, scambiare o acquistare tutto ciò che più vi sta a cuore.
Cosa ne dite? Un programma davvero eccitante, ricco di sorprese e colpi di scena.
Che aspettate quindi? "Agguantate" rivista e cassetta e... largo ai giochi.
Buon divertimento e... arrivederci al mese prossimo.
Ciao!

I numeri arretrati di LINGUAGGIO MACCHINA e LINGUAGGIO MACCHINA 16/20 possono essere richiesti direttamente alla nostra redazione al seguente indirizzo:
Edizioni Foglia - Via Jenner 31 - 20100 Milano allegando un vaglia postale (appogg. a poste VR-MI) o un assegno di 12.000 lire
I numeri arretrati di TUTTOCOMPUTER verranno invece inviati contrassegno.

cosa contiene la cassetta?

COMMODORE 16:

1. **Keys**
2. **Galaxy**
3. **Olympic Games I**
4. **Olympic Games II**
5. **Tombola**

MSX:

1. **Birdman**
2. **Dot Assembler**
3. **Death River**
4. **Sound Maker**
5. **Invasion**

GAMES GAMES GAMES
GAMES GAMES GAMES

attenzione! attention! look out! achtung!
non perdete questa pagina!
attenzione! attention! look out! achtung!

Occhio all'azimut

Per la buona lettura della cassetta occorre che la testina del registratore sia pulita ed allineata col nastro. Se così non fosse potrebbe accadere che sul video appaia "error". Pulite allora la testina del registratore con un cottonfioc imbevuto di alcool. Se nonnostante questa operazione il computer continua a non caricare bene prendete un cacciavite ed agite direttamente (in senso orario o antiorario) sulla vite apposita per la regolazione dell'azimut.

Testina fuori azimut (agire in senso orario)

Testina fuori azimut (agire in senso antiorario)

Testina perfettamente allineata

Nastro

Nastro

MODALITÀ DI CARICAMENTO

Commodoristi: digitate LOAD + RETURN ed avviate il registratore. Al termine di ogni caricamento spegnete il registratore.
Se possedete un MSX, per il caricamento di ogni singolo programma seguite le indicazioni riportate alla rubrica MSX Challenge. Per farlo partire digitate Run e Return oppure F5.
Per un buon caricamento dei programmi è opportuno tenere il registratore lontano dal monitor e dall'alimentatore.

Per facilitarvi il ritrovamento dei giochi sulla cassetta vi consigliamo di scrivere il minutaggio di ciascun programma accanto ai titoli prestampati che troverete all'interno della fascetta.

CAMPIONI DI TUTTO IL MONDO SFIDATEVI NELLA NOSTRA

La competizione è l'anima del gioco ma, il gioco, per essere ancora più avvincente, ha bisogno di un premio che gratifichi lo sforzo compiuto.
Allora, cari amici, che ne dite di una **gara** tra voi lettori? Come fare?
Semplice: compilate il modulo allegato ed inviatelo alla nostra redazione precisando il nome del gioco, scelto tra quelli contenuti mensilmente sulle nostre cassette, ed il punteggio ottenuto.
Tutti i mesi proclameremo i campioni delle specialità. La nostra classifica verrà aggiornata ad ogni numero perciò, se volete cimentarvi in più giochi, se volete migliorare un record, se volete che il vostro nome troneggi in cima alle alte vette della HIT-GAMES in più giochi, beh!, sta a voi... Per comprovare il record ricordatevi di inviarci una vostra polaroid accanto allo schermo del computer col punteggio ottenuto: allora non avremmo proprio più dubbi nel proclamarvi campioni del mese e la vostra foto verrà pubblicata sulle pagine della rivista.

NOME .... COGNOME ....
INDIRIZZO .... CITTÀ ....
ETÀ .... TITOLO DEL GIOCO ....
RECORD ....

il mercatino dei lettori

***Vendo/Scambio** giochi per C64 a L. 2.000, un affare!!! STEFANO ROSSI - Via C. Colombo, 3 - 30016 Jesolo (Venezia) - Tel. 0421/951213.

***Vendo/Scambio** software per C64 su nastro. Siamo un Club di amici e possediamo le ultime novità riguardo giochi ed utilities. Rispondiamo a tutti. Prezzi incredibili. VALERIO CUCCHI - Via dei Gonzaga, 94 - Reggio Emilia - Tel. 0522/53721.

***Cerco** stampante Plotter della Commodore a prezzo che non superi 150.000, cerco anche Disk-drive Commodore 1541 a prezzo che non superi L. 250.000, cerco anche cartucce per C16 a L. 20.000. Annuncio sempre valido. Telefonare dopo le ore 21. GIOVANNI PARRINO - Via Guardi, 50 - 20092 Cinisello Balsamo (MI) - Tel. 02/6171190.

***Vendo** videogiochi (di tutti i tipi) ed utilities per i computer: C64, Vic-20, PLUS-4, C16, ZX Spectrum a prezzi che vanno da L. 1000 a L. 5.000, oppure scambio con materiale di qualsiasi tipo per C16. Telefonare dopo le 21 per avere la lista. Annuncio sempre valido. GIOVANNI PERRINO - Via Guardi, 50 - 20092 Cinisello Balsamo (MI) - Tel. 02/6171190.

***Vendo/Scambio** giochi ed utilities per C64 vasto assortimento, possiedo più di 400 titoli su disco e cassetta, massima serietà. SALVATORE LAZZARO - Via Argo, 3 - 91016 Bonagia Erice (Trapani).

***Vendo** cassette per C64 con 15 giochi a L. 20.000 o singoli a L. 1.500, tutte novità. Inoltre vendo Commodore VIC 20 + Registratore dedicato + Libro VIC 20 + 15 giochi + 1 cassetta di introduzione al computer + 1 cassetta con Video Agenda degli amici a L. 250.000. MASSIMILIANO TRADORI - Via Mascagni 74 - 09045 Quartu S. Elena (CA) - Tel. 070/815296.

***Vendo** Spectrum 48 K con alimentatore, Interfacce programmabili, Joystick, cassetta introduttiva per lo Spectrum ZX, cassetta archivio dati, 2 libri con programmi e 150 giochi a L. 400.000 trattabili. Scrivere a: FRANCESCO MANZO - Villaggio Azzurro, 4/C - 29019 S. Giorgio Piacentino (PC).

***Vendo/Scambio** giochi per C64 (Bruce Lee, Impossibile Mission, Sala giochi). Ogni gioco a L. 3.000 trattabili. STEFANO LAGUZZI - Via Carpeneto 22 - 15070 Castelferro (AL) - Tel. 0131/710131.

***Vendo** ultime novità per il C64 a prezzi bassissimi. Scrivete a: MARIO SCHIANO - Viale Kennedy, 425 - 80125 Napoli.

***Il Club Spectromania** vende 12 programmi tra i migliori come Hyper Sports ed altri, a sole L. 30.000. Per informazioni: CLUB SPECTROMANIA - Via delle Romite, 8 - 50125 Galluzzo (FI) - Tel. 055/2048905.

***Vendo/Scambio** programmi C64 e utilities (Copy 190, Simon's Basic, Turbo Tape, Super Turbo Tape) a L. 1.000 cad. FEDERICO MILANI - Via Càprera, 37 - 10136 Torino - Tel. 011/353416.

***Cerco videogames** Basket, Tarzan, Karatè Kid per C16; **vendo** fantastici games per C16, C64 e VIC 20; **scambio** video games per C16. PIERLUIGI COVELLI - Via S. Pietro, 128 - 72027 S. Pietro Vernotico (Brescia) - Tel. 0831/653044.

***Cerco** soci per un club di prodotti Commodore (C64, VIC 20, C16, PLUS/4) per iscriversi mandare L. 3.500, per avere diritto all'accesso alla nostra biblioteca di software, la tessera, un giornale... scrivere a: GIUSEPPE GUGLIELMINO c/o HC Club 112 - Via M. Scammacca, 18 - 95127 Catania - Tel. 095/372097 - 377183.

***Vendo** cassette C60 con minimo 50 giochi (con omaggio di cassetta fornita di Simon's Basic e Turtle Graphic) per CBM 64 a L. 70.000. Oppure giochi a scelta a L. 1.500 cadauno. Telefonare ore pasti. MATTEO SELLITRI - Vico I Ortale S. Domenico, 19 - 71042 Cerignola (Foggia) - Tel. 0885/27473.

***Vendiamo** giochi ed utilities per C64 a prezzi stracciati. CLUB R.P.R.64 - Via Ampere, 1 - 24047 Treviglio (Bergamo) - Tel. 0363/44665.

***Vendo** giochi per C64 a L. 1.000 cadauno. Posseggo inoltre 300 titoli novità. MAURO NEPOTE - Corso Nazioni Unite, 86 - 10073 Ciriè (TO) - Tel. 011/9209023.

***Vendo** programmi per C64 da L. 1.500 l'uno. ANTONIO TONCELLI - Via Maggi, 76 - 57100 Livorno - Tel. 0586/33228.

***Vendo/Scambio** videogames novità. Tutti i migliori giochi a prezzi stracciati per C64. Offro e chiedo massima serietà. Inviare francobolli per risposta sicura. FRANCESCO GRANAI - 53041 Loc. Grottoli (Siena) - Tel. 0577/712306.

***Vendo** per C64 fantastici videogames prezzi stracciati. Regalo il supporto e spese postali. Lista gratuita a tutti. VALENTINO BAIO - Casella Postale, 29 - 36016 Thiene (Vicenza).

***Vendo/Scambio** ottimi giochi per C64 a L. 1.000 cadauno. FRANCO CROCETTA - Via Orsolone Guantai, 38/1 - 80131 Napoli - Tel. 081/466539.

***Vendo** C64 + drive + registratore + Joystick + 150 giochi su disco + 150 giochi su cassetta con portadischi a L. 800.000. Un affare. SILVIO GATTA - Via L. Calori, 1 - 40122 Bologna - Tel. 051/556218.

***Vendo/Scambio** giochi per C64 a L. 800 cadauno. Telefonare ore pasti. FLAVIO CENTOSCUDI - Via Pievaiola, 202/R - Castel del Piano (PG) - Tel. 075/7740230.

***Vendo** cassette con programmi per C64, Spectrum e Vic 20 (acquistate in edicola) mai usate. Vendo inoltre libri e riviste, vero affare. GIORGIO SASDELLI - Via Vespri Siciliani, 31 - 20146 Milano - Tel. 02/4238485.

***Vendo/Scambio/Acquisto** software di ogni tipo per C64. Dispongo di tutte le ultime novità da U.S.A. Inghilterra e Germania. Prezzi alla portata di tutti. Invio lista gratuita. PAOLO UCCELLATORI - Via Vandalino, 138 - 10142 Torino - Tel. 011/792574.

***Vendo** giochi per C64 a prezzi variabili. Cerco inoltre gioco Hunch Back 2 da scambiare con uno dei miei. DAVIDE MOSCA - ANDREA SEGA - Via Ghemme, 27 - Via Gravere, 31 - 10145 Torino - Tel. 011/759560 - 011/7490580.

***Vendo/Scambio** software per C64, scambio solo con programmi del valore di Tot Professional, Winter Games, Empire ecc. MAURIZIO PORZIONATO - Via A. Avogadro, 22 - 13010 Quaregna (VC) - Tel. 015/94751.

***Vendo/Scambio** giochi per C64 su cassetta con 42 giochi e programmi a L. 15.000, e una cartridge Comega rage a L. 13.000 telefonare ore pasti, a chi compra più di 20 giochi 5 giochi in regalo a partire da L. 4.000 l'uno. FABRIZIO COCCIA - Via E. Toti, 7 - 00019 Tivoli (Roma) -Tel. 06/26100.

***Vendo/Scambio** programmi per C64 su disco, ultime novità. MAURIZIO PORZIONATO - Via Amedeo Avogadro - 13010 Quaregna (Vercelli) - Tel. 015/94751.

***Vendo** circa 200 giochi con turbotape per C64, per avere la lista basta telefonare ore pasti. DARIO MORGANTE - Via Etruria, 53 - 00058 S. Marinella - Tel. 0766/737276.

***Vendo/Scambio** un centinaio di giochi di buon livello per il C64 (L. 2.500 cadauno). Vendo e scambio anche cassette acquistate in edicola. Solo zona Milano. GIORGIO SASDELLI - Via Vespri Siciliani, 31 - 20146 Milano - Tel. 02/4238485.

***Vendo** programmi per C64. Ultimi americani ed inglesi. Disco o cassetta. Solo Roma.

STEFANO SOLLA - Monsummand Terme, 7 - 00148 Roma - Tel. 06/5258377 - 536006.
*Scambio giochi per C64 solo su nastro. Ne possiedo di ottimi.
STEFANO CROSARA - Via A. Gramsci, 228 - 57100 Livorno - Tel. 0586/855791.
*Vendo/Scambio giochi in linguaggio macchina per C64. I migliori esistenti sul mercato. Prezzi irrisori. Richiedere lista o spedire la propria.
ROBERTO FERRARO - Vico Buono, 7 - 80069 Vico Equense (Napoli) - Tel. 081/8799937.
*Vendo programmi per C64 a prezzo di regalo.
GIANPAOLO DE MARCHI - Via G. Galilei, 1 - 10080 Bosconero (TO) - Tel. 011/9889002.
*Vendo software per C64 a prezzi fra L. 2.000 e 10.000 + spese postali. Inviate L. 1.000 per ricevere la lista.
STEFANO FALANGA - Via Ignazio Sorrentino, 29/F - 80059 Torre del Greco (NA) - Tel. 081/7710534.
*Vendo C16 praticamente nuovo con registratore, joystick, istruzioni in italiano e circa 50 programmi e listati a L. 350.000. Ancora in garanzia. Telefonare ore pasti. Prezzo trattabile.
LUCA GUARNERO - Via Stoppele, 17 - 37133 Verona - Tel. 045/533063.
*Cerco disperatamente gioco del Karatè. Compro o scambio con gioco a scelta. Telefonare ore pasti.
ALESSANDRO SCALAS - Via Argonne, 9 - 09100 Cagliari - Tel. 070/290328.
*Vendo giochi su nastro per C64 a sole L. 250 l'uno + cassetta + spese postali. A chi ne fa richiesta invio lista giochi.
EDOARDO CITTERIO - Via Trento Trieste, 80 - 20046 Biassono (MI) - Tel. 039/753461.
*Vendo/Scambio software per C64 su disco e cassetta. Ottimi prezzi.
ADRIANO NIEDDU - Corso Moncalieri, 1 - Torino - Tel. 011/888154.
*Vendo/Scambio tantissimi videogames e utilities per C64 a prezzi ridicoli (max L. 4.000).
UGO FESTINI - Via XXV Aprile, 8 - 22059 Robbiate (CO) - Tel. 039/510604.
*Vendo Atari computer 800XL + 7 giochi + manuale istruzioni + 1 Joystick + 2 cassette Basic + libro programmazione come nuovo a L. 350.000 trattabili.
GIORDANO SEMENZIATO - Via Certosa, 19 - 20038 Seregno (MI) - Tel. 0362/235106.
*Vendo giochi su nastro per C64 a L. 1.000. Inoltre cerco il Karatè per C64.
GIANLUCA MURA - Via Caravaggio, 47 - 07041 Alghero (Sassari) - Tel. 079/952582.
*Cerco/Scambio cassette del C64 (giochi). L'annuncio è rivolto solo alle persone di Livorno e provincia.
MASSIMO MESSINA - Borgo Cappuccini, 98 (Livorno) - Tel. 0586/31695.
*Vendo Atari 2006 con 5 cassette gioco. Il tutto a L. 150.000.
ALDO DELLA SANTA - 3ª traversa - 80126 S. Donato (Napoli) - Tel. 081/7263120.
*Vendo per C64 cassetta con 40 giochi in turbo a L. 13.000 + spese postali.
FILIPPO BALDELLA - Via Ovidio 9/8 - 63039 S. Benedetto del Tronto (Ascoli Piceno) - Tel. 0735/85750.
*Vendo programmi per C64 anche all'estero, tutte novità, prezzi bassissimi.
MAURIZIO COTONI - Via Masaccio, 92 - 50132 Firenze - Tel. 055/240798.
*Vendo cassetta per C64 con 50 giochi nuovissimi a L. 25.000. Inoltre vendo turbo di otto giri a L. 4.000.
RAFFAELLO SICILIANO - Via della Quercia, 3 - 84080 Pellezzanno (SA) - Tel. 089/271071.

1	SCARABEUS	ARIOLASOFT	CBM 64
2	THE GOONIES	DATASOFT	CBM 64
3	WINTER GAMES	EPYX/US GOLD	CBM 64
4	RAMBO	OCEAN	CMB 64
5	STARQUAKE	BUBBLE BUS	ZX SPECTRUM
6	ZORRO	DATASOFT	CBM 64
7	LITTLE COMPUTER PEOPLE	ACTIVISION	CBM 64
8	FIGHTING WARRIOR	MELBOURNE HOUSE	CBM 64/ ZX SPECTRUM
9	MONTY ON THE RUN	GREMLIN GRAPHICS	ZX SPECTRUM
10	IMPOSSIBLE MISSION	EPYX/US GOLD	ZX SPECTRUM

 nuova entrata

L'ASSEMBLER PER MSX

prima lezione
di Massimo Cellini

A partire da questo numero inizieremo un corso di assembler che vi permetterà, in breve tempo, di programmare tramite questo linguaggio, mettendendovi quindi a disposizione tutte le reali potenzialità del vostro MSX.
Sarà bene comunque chiarire innanzitutto **cosa è l'assembler** *e quali differenze presenta rispetto ad altri linguaggi più noti (BASIC, Pascal, ecc.).*
Per prima cosa dovrete sapere che il vostro computer è in grado di capire solamente un linguaggio molto elementare, dove ogni istruzione è rappresentata da un numero o una sequenza di numeri binari (all'interno del computer tutto è rappresentato in binario).
Questo linguaggio è tipico di ogni **CPU** *(la CPU è il "cervello" del computer) e si chiama appunto linguaggio macchina (l/m).*

Per far si che il computer "capisca" altri linguaggi più semplici occorre un apposito programma scritto in l/m che traduca ogni istruzione di quel linguaggio in una serie di istruzioni in l/m; un tale programma si chiama **interprete.** *Un tipico esempio può essere il BASIC, un linguaggio sicuramente semplice in quanto si avvicina molto come struttura alla lingua parlata da noi, ma è decisamente incomprensibile per la CPU. Un linguaggio semplice che usa istruzioni derivate dalla nostra lingua e permette operazioni complesse si chiama linguaggio ad alto livello; fra i molti linguaggi ad alto livello possiamo ricordare il BASIC, Pascal, Logo, COBOL, APL, LISP ecc. Tutti questi, benchè radicalmente diversi fra loro, presentano le caratteristiche descritte in precedenza e tutti necessitano di appositi interpreti per essere compresi dal computer.*
Fra i linguaggi ad alto livello possiamo poi distinguere due categorie: **i linguaggi interpretati ed i linguaggi compilati.**

Qual è la differenza? *Ogni volta che noi eseguiamo un programma scritto in un linguaggio interpretato, ogni istruzione di quel programma deve essere ritradotta nel suo equivalente in l/m con un evidente dispendio di tempo, mentre i linguaggi compilati vengono interpretati una sola volta, dopodiché il programma viene trasformato nel suo alter-ego in linguaggio macchina cosicché, quando lo farete "girare" successivamente, esso non avrà più bisogno di essere tradotto in quanto è già stato interamente convertito in l/m. Purtroppo questo sistema presenta degli svantaggi, poiché, se noi volessimo apportare delle modifiche al programma dopo averlo compilato, non potremmo più farlo a meno di non aver preventivamente salvato su un supporto di massa il codice sorgente del programma stesso. Spesso però l'operazione risulta lunga e scomoda, specialmente in fase di messa a punto di programmi molto complessi. Per questo motivo la stragrande maggioranza dei personal computer implementa un interprete BASIC il quale, oltre a essere un linguaggio molto semplice ed utilizzabile per svariate applicazioni, ha il vantaggio di essere interpretato e, quindi, è possibile, in ogni momento, modificare il programma. Naturalmente questa comodità viene penalizzata con una bassa velocità di esecuzione, ma nella maggior parte dei casi i benefici superano di gran lunga gli svantaggi derivanti dall'uso di un linguaggio interpretato.*

Anche nel caso di linguaggi compilati la velocità di esecuzione del programma non eguaglia certo quella che si può ottenere scrivendo il programma direttamente in l/m, questo perché, usando un linguaggio ad alto livello, bisogna spesso seguire delle strade più lunghe del necessario per raggiungere lo scopo mentre in l/m si può prendere una "scorciatoia", arrivando direttamente al punto e eseguendo solo le operazioni strettamente necessarie alla riuscita del programma.
Tornando al discorso fatto precedentemente possiamo concludere che il linguaggio macchina è composto da istruzioni molto semplici che vengono eseguite in tempi piccolissimi, mentre i linguaggi ad alto livello sono composti da istruzioni generalmente molto complesse che richiedono un tempo relativamente lungo per venire eseguite.

Da ciò si deduce che il linguaggio macchina è indispensabile in tutti i casi in cui sia determinante la velocità di esecuzione, al di là della complessità del programma.

L'ASSEMBLER

Ricordate quando abbiamo detto che il linguaggio macchina è costituito da sequenze di numeri? Vi renderete certamente conto che è impensabile programmare scrivendo uno o più numeri per ogni istruzione che si desidera comunicare alla CPU, anche perché sarebbe davvero una impresa ardua ricordarsi tutti i codici operativi nelle numerose istruzioni eseguibili in l/m. Per questo motivo sono nati i **linguaggi assembler**.

In pratica il linguaggio assembler è costituito da determinati codici mnemonici a ognuno dei quali è associata una determinata istruzione in linguaggio macchina (diversamente dai linguaggi compilati ad alto livello dove, ad ogni istruzione, corrispondevano centinaia di istruzioni in l/m).

Nel linguaggio assembler esiste quindi un rapporto di uno a uno fra le istruzioni ed i codici operativi del linguaggio macchina. Possiamo dunque affermare che, ai fini della programmazione, **l'assembler è equivalente al linguaggio macchina**.

Per meglio chiarire quanto esposto vi proponiamo un semplice esempio. **Se il codice operativo di un'istruzione per incrementare il contenuto di un registro** *(spiegheremo in seguito cosa sono i registri)* **è il seguente:**

00111100

in assembler esso diventa:

INC A

che è poi la contrazione di INCREASE A, cioè INCREMENTA A, dove A è un registro della CPU, che per il momento potete immaginare come una variabile del BASIC.

Purtroppo la CPU non è assolutamente in grado di capire l'assembler. Occorre quindi, anche in questo caso, un determinato programma, chiamato **assemblatore**, *il quale si occupi di compilare il programma assembler associando ad ogni istruzione un determinato codice del linguaggio macchina. Il programma diviene così direttamente interpretabile ed eseguibile dalla CPU.*

Naturalmente, dopo essere stato assemblato, il programma viene posto in memoria sotto forma di numeri e quindi sarebbe di ben difficile interpretazione da parte di un'altra persona interessata od anche dallo stesso programmatore che voglia rivederlo a distanza di tempo; per questo esistono appositi programmi che eseguono il compito inverso degli assemblatori, convertono cioè il linguaggio macchina nelle rispettive istruzioni assembler. Questi programmi si chiamano **disassemblatori**.

Non tutte le istruzioni in l/m sono comunque formate da un solo numero: alcune arrivano ad occupare perfino sequenze di quattro numeri a otto bit (tratteremo più approfonditamente l'argomento quando ci occuperemo del binario), quindi il codice assembler dipende dall'ordine in cui sono letti e interpretati i codici del linguaggio macchina in quanto, iniziando a disassemblare il programma da un punto sbagliato, falseremmo completamente il risultato. Ma non preoccupatevi ora, lo capirete più chiaramente quando inizieremo a parlare dell'assembler vero e proprio.

Bene, per ora terminiamo qui il nostro nuovo corso di assembler MSX, sperando che abbiate capito senza difficoltà i concetti esposti in queste pagine, e vi rimandiamo alla prossima puntata per avere un quadro completo di come vengono trattati i numeri in **binario** *e in* **esadecimale: due sistemi di numerazione fondamentali per la programmazione in assembler.**

sfida al co

1. KEYS

Ti ritrovi a doverti destreggiare in un immenso labirinto, disseminato di svariati oggetti misteriosi, composto da ben 21 stanze.
Scopo del gioco è quello di entrare in possesso delle chiavi nascoste nel labirinto per poter, finalmente, guadagnare l'uscita.
La difficoltà maggiore consiste nella presenza costante di una serie di elementi di disturbo: sono degli omini e dei piccoli rombi colorati che, muovendosi in tutte le direzioni, aggirandoti e inseguendoti, ti intralceranno di continuo il cammino per impedirti l'accesso alla cassaforte.
Però, con il tuo raggio laser, hai la possibilità di colpirli per aprirti un varco fra di loro.
Gli omini sono più vulnerabili perché, nel caso che tu li colpisca, moriranno e spariranno dallo schermo. I rombi, invece, verranno solo momentaneamente "narcotizzati" dai tuoi spari, per riprendere subito dopo la loro azione di disturbo. Entrambe i "disturbatori" andranno comunque evitati accuratamente perché possono causare guai seri alla tua... salute!
Ma non è tutto: vedrai infatti una serie di quadrati colorati che fanno da cinta allo scrigno che contiene la chiave che dovrai recuperare. Con l'aiuto del tuo raggio luminoso potrai aprirti un varco per andare a sovrapporti esattamente sulla chiave, così da poterne entrare in possesso.
Forza, la strada è lunga, le chiavi sono tante ma... la libertà val bene un po' di fatica!
5 livelli di gioco con difficoltà crescenti.

TASTI:
Joystick in porta 2

Joystick:		*Tastiera:*
	↑ su	*W*
	↓ giù	*X*
	← sinistra	*J*
	→ destra	*:*
FIRE	*fuoco*	*barra spazio + X*

2. GALAXY

Ci troviamo in presenza di un ennesimo attacco di invasori spaziali che, non soddisfatti di aver fiaccato le nostre forze nel corso dei numerosi precedenti attacchi, sono convinti di poter sferrare un'ultima offensiva e di impadronirsi definitivamente del nostro pianeta.
Ma questo non sarà affatto facile e, comunque, non è ancora detta l'ultima parola, dato che abbiamo messo a punto nuove armi e sorprendenti trabocchetti che, molto probabilmente, renderanno vano ogni sforzo e costringeranno i nemici a battere in ritirata.
Certo le astronavi aliene sono molte e pericolosamente equipaggiate. Ma le nostre possibilità non sono di certo da sottovalutarsi e visto che, finora, siamo riusciti a scacciarli, potremo in ogni caso tentare una controffensiva.
Lo scopo l'hai capito: contrattaccare massicciamente e cercare di evitare i colpi nemici ad ogni costo.
Coraggio! Fagli capire (e vedere!) chi siamo.

TASTI:

Barra spazio	*= per giocare*
Shift	*= per sparare*
Z	*= sinistra*
X	*= destra*

3. OLYMPIC GAMES 1
4. OLYMPIC GAMES 2

Finalmente ci siamo! Quattro anni sono lunghi da passare ma, adesso, è arrivato il momento delle tanto attese Olimpiadi.
Per te questi anni sono stati di duro lavoro: diete ferree, ore ed ore di intensi allenamenti per prepararti a gareggiare al meglio nelle varie discipline: dal salto in lungo ai 100 metri piani, dal lancio del giavellotto ai 110 metri ostacoli, dal lancio del peso al salto in alto.
Hai stretto i denti per vincere la stanchezza che, tante volte, ti attanagliava ma ora, con orgoglio, puoi dire che i tuoi sforzi sono stati ricompensati.
La tua squadra ha puntato tutto su di te ed il pubblico ti attende come la nuova promessa dello sport. Non puoi deluderli!
I tuoi avversari sono altrettanto forti e decisi a vincere, ma tu puoi farcela. Non accontentarti di una semplice qualificazione ma tenta il tutto per tutto per battere il record mondiale.
Quando il gioco comparirà sul tuo schermo, avrai subito la piacevole sensazione di trovarti in uno stadio olimpico: vedrai che colori, che folla assiepata sugli spalti e... che simpatico atleta!
I due giochi che ti presentiamo sono quelli delle Olimpiadi e il divertimento è assicurato. Non arrabbiarti troppo se perdi ma, anzi, rimettiti subito in pista per migliorare la tua posizione in classifica.
Le gare sono molte e varie: corsa, corsa ad ostacoli, lancio del peso... Vedrai nei vari livelli cosa succederà... Ma, lasciamo a te la piacevole scoperta!
Per qualificarti dovrai ottenere dei buoni punteggi altrimenti verrai penalizzato con il ritiro dalla gara e, questo, un campione della tua stoffa non può proprio permetterselo!
Tre tentativi a tua disposizione per ogni disciplina che affronti.

mmodore

TASTI:

OLYMPIC GAMES 1:

Joystick in porta 2
Joystick verso sinistra/destra = per correre
FIRE = per saltare

OLYMPIC GAMES 2:

J o K = per selezionare il gioco con Joystick o Tastiera
Joystick in porta 2

Joystick:		*Tastiera:*
Joystick	*per correre*	*Q e Y*
FIRE	*per saltare*	*Barra spazio*

5. TOMBOLA

Eccoci qui con un gioco facile facile e vecchio quanto il mondo (almeno si crede!) ma che rispunta fuori appena ci si ritrova con un po' di amici.
Ecco per voi una Tombola computerizzata che non ti risparmierà i soliti fagioli secchi (quando mancano le pedine!) ma ti eviterà di frugare nel sacchettino alla ricerca dei numeri.
Puoi giocare con tanti amici o da solo con il computer: a te la scelta!
La tastiera ti aiuterà ad estrarre i numeri, che compariranno sul tuo schermo nella parte superiore della tabella, e che verranno evidenziati nella casellina corrispondente per contrasto luminoso.
Le regole sono quelle di sempre. I premi per le terne, le quaterne e le cinquine, stabiliscili tu.
Allora buon divertimento a tutti con questa super tombola computerizzata!

TASTI:
Barra spazio = per far uscire un nuovo numero
F = per terminare la partita e cominciarne una nuova

N.B. PER IL CARICAMENTO DI OGNI SINGOLO GIOCO DIGITARE: LOAD"",1,1 e RETURN

MSXch

1. BIRDMAN

Ti sei cacciato in un brutto pasticcio, ti sei offerto volontario per salvare una principessa che si trova nei guai. Entri in una specie di labirinto ed il tuo scopo è quello di recuperare una chiave ed un anello e quindi di correre con questi oggetti a salvare la principessa che ti aspetta a braccia aperte.
Durante il tragitto ti troverai di fronte a delle porte chiuse e ad altre aperte. Attento: sii abile nell'aprirle e nel chiuderle perché saranno la tua via di scampo quando verrai inseguito dai nemici.
Per aprire le porte dovrai posizionarti sulla freccia OPEN, per chiuderle sulla freccia CLOSE.
I nemici sono diversi, e tu devi fare molta attenzione alla tua riserva di energia! Stai lontano, quindi, dai rombi in movimento e dagli alieni perché se li tocchi, perderai forza.
Alla base del labirinto, ci sono anche delle enormi vesciche di acido: attento a non fare un tuffo altrimenti la tua energia si esaurirà definitivamente.
Per fortuna, però, nel labirinto ci sono anche degli amici; cerca le stelle e le bandiere ogni volta che vorrai rifornirti di energia.
Forza, è ora di correre a cercare i due oggetti, la principessa ti aspetta.
Vari i livelli di difficoltà con crescenti pericoli.
Per caricare il gioco digitate: BLOAD "CAS : bird", R.

TASTI:
Joystick in porta 1 o 2

Joystick:		*Tastiera:*
FIRE	*per giocare e per volare*	*Barra spazio*
	per muoversi tasti cursori	

2. DOT ASSEMBLER

Il primo programma di utilità che vi forniamo in questo numero è un assemblatore.
Vi sarà utilissimo per programmare in assembler quando vi troverete a dover affrontare più da vicino la programmazione in linguaggio macchina.
Tenete, dunque, questo "DOT" ben a portata di mano perché, una volta imparato l'uso, non potrete più farne a meno!
Tra i numerosissimi comandi a disposizione (comandi che, lo ripetiamo, vi torneranno quanto mai utili man mano che progredirete nello studio dell'assembler) troviamo:

E = permette di scrivere un programma in assembler
A = assembla un programma scritto in assembler e lo trasforma in linguaggio macchina
P = permette di ottenere un listato del programma, che è stato scritto precedentemente, in assembler
X = mostra tutti i registri della CPU e i relativi valori correnti
u = disassembla una zona di memoria

Gli altri comandi sono molto complessi e verranno trattati e discussi nel corso delle lezioni "L'ASSEMBLER PER MSX".
Per caricare il programma digitate: BLOAD "CAS : dot", R.

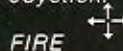
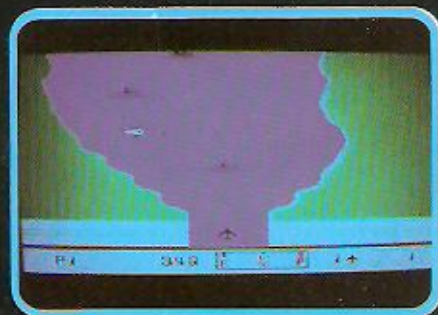
TASTI:
Tastiera

3. DEATH RIVER

Siediti e preparati ad un bellissimo gioco, molto particolare, che ha colpito anche me quando l'ho visto.
Carica il tuo computer come sai e premi un tasto qualsiasi: comparirà una bellissima distesa verde con un corso d'acqua blu che lo attraversa longitudinalmente.
Tu sei al comando di un piccolo aereo e il tuo compito è quello di portarlo in salvo. Nel corso del tuo volo dovrai scansare i numerosi ostacoli che si ergono al tuo passaggio, evitando altresì di uscire dalla rotta che ti viene indicata dal tracciato del fiume, anse comprese.
Parti pure tranquillo e, davanti ai tuoi occhi, vedrai comparire le cose più incredibili: palloni aerostatici, elicotteri o navi (che dovrai accuratamente scansare!), aerei e carri armati che, però, ti spareranno addosso senza pietà.
Ricordati una cosa molto importante: il tuo velivolo può scoppiare da un momento all'altro se, ad esempio, sorvoli la terra ferma, se vai ad urtare contro qualche ostacolo o se esaurisci il carburante.
Per evitare questa evenienza è stato predisposto un segnale acustico che funzionerà da allarme.
Inutile dire che l'esplosione avverrà in ogni caso se sarai colpito dagli altri aerei o dai carri armati.
A mano a mano che procederai nel percorso, ad un certo punto ti troverai in una strettoia, in corrispondenza ci sarà un ponte che delimita ogni livello di difficoltà del gioco.
Chiaramente non sarà così semplice arrivarci ed andare oltre perché, proprio qui, dovrai destreggiarti fra mille inconvenienti, sorprese e diavolerie.
Ma non ti spaventare: non perdere la calma e tieniti pronto a tutto perché, più avanti procederai, più aumenteranno le imboscate, gli ostacoli, gli attacchi e la frequenza con cui ti si presenteranno.
Fai attenzione ai simboli sullo schermo, potrai infatti verificare continuamente il livello del carburante, oltre al punteggio ed alla situazione delle tue vite...
Scopo del gioco è di arrivare alla foce del fiume integri.
Un consiglio: vai veloce, spara il più possibile, non perdere occasione per fare un abbondante rifornimento di carburante e... vedrai che con un po' di impegno ce la farai senz'altro.
Tre vite a disposizione. Bonus vitae ogni 10.000 punti.
Per caricare il gioco digitate: BLOAD "CAS : death", R.

allenge

TASTI:
Joystick in porta 1 o 2

Un tasto qualsiasi per giocare (con joystick o tastiera)
premendo CTRL + STOP durante gioco si ritorna al Menu
premendo STOP si elimina il sonoro
premendo STOP per due volte torna il sonoro
premendo ESC si ferma il gioco
premendo ESC per due volte riprende il sonoro

Joystick:		*Tastiera:*
✥	*per muoversi*	*tasti cursori*
FIRE	*per sparare*	*barra spazio*

4. SOUND MAKER

Il secondo programma di utilità che ti presentiamo è un potente generatore di effetti sonori.
Il computer dispone di tre voci (di suono o rumore bianco) che possono o meno essere emesse in contemporanea.
Sullo schermo potrete vedere lo schema di funzionamento.
Che cosa dovete fare con tutto ciò? Tramite la pressione di alcuni tasti, si potranno alterare i parametri di definizione del suono: frequenza, volume, inviluppo e forma d'onda.
Vediamo subito i due generatori che sono sintetizzati con R, per il rumore e con S per il suono.
Questi potranno essere aperti o chiusi (assenza di emissione sonora). I simboli ovali corrispondenti si coloreranno di nero in assenza di emissione sonora, viceversa si apriranno a grata. Partiamo subito dal primo punto e parliamo della frequenza.
Per il suono ne abbiamo tre e vengono utilizzate per generarlo, modificarlo o definirlo.
Per il rumore ne abbiamo, invece, una sola per tutti e tre i generatori.
Per quanto concerne il volume abbiamo tre indicatori che segnalano la variazione d'intensità da zero al massimo. Se il cursore è a fondo scala, il volume segna la variazione, determinata dalla forma d'onda presente in quel momento. Viceversa il suono viene messo in volume costante.
Terzo punto: l'inviluppo, che definisce il tempo impiegato dal suono per passare dal volume zero al volume massimo o viceversa. Può essere variato, quindi, a seconda della forma di onda che si usa.
Quarto punto: la forma d'onda o andamento assunto dall'emissione sonora, con otto possibilità di scelta. Può essere quindi selezionata tramite il cursore e a seconda dei casi può presentare un andamento ciclico o meno.
Per caricare il programma digitate: CLOAD "sound" : RUN

TASTI:
Con i tasti cursori ci si sposta sul parametro desiderato:
alto = indietro
basso = avanti

Con i tasti cursori si altera il parametro:
destro = apre
sinistro = chiude

SOUND MAKER

F1 = si ascolta il suono attuale che si è programmato
F2 = interrompe se c'è emissione sonora
F3 = vengono visualizzati i registri del PSG interessati alla generazione dei suoni ed i relativi valori per generare il suono da programmare in quel momento.

Premere un tasto qualsiasi per far ritorno allo schermo operativo iniziale.

5. INVASION

Mettiti comodo, rilassati e concentrati. Una nuova impegnativa missione ti attende. Dagli spazi interstellari più lontani e sconosciuti è partita una flotta di cinque astronavi aliene che si sono messe sulle tue tracce. Non appena ti avranno individuato, cercheranno in tutti i modi di distruggerti aggirandoti, sparandoti e rincorrendoti.
Tu sei solo ma la tua possibilità di contrattacco è altrettanto incisiva di quella degli invasori.
Una cosa importante di cui devi tener conto è che hai un quantitativo limitato di energia disponibile, che diminuirà ad ogni tuo attacco.
Fai attenzione, però, perché se sarai tu a subire l'attacco, la tua scorta di energia avrà un calo ancora maggiore. (almeno del doppio!!)
Quindi, cerca di evitare a tutti i costi di rimanere sotto il tiro dei tuoi avversari e mettiti invece in posizione per poterli colpire per primo.
Superando i combattimenti potrai via via aumentare il grado di difficoltà del gioco, passando ad un livello superiore, cosa che implicherà una maggiore frequenza di incursioni nemiche.
Tieni presente, però, che le unità spaziali si muoveranno con una velocità crescente.
La tua abilità e i tuoi riflessi sono quindi elementi essenziali per arrivare a distruggere gli alieni che, in ogni caso, non suoneranno mai la ritirata e non scompariranno dal tuo schermo finché... non li avrai uccisi tutti.
Impegnati quindi al massimo.
Forza e... in bocca all'alieno!!!
Per caricare il gioco digitate: CLOAD "inv" : RUN

TASTI:
Barra spazio = fuoco
Cursori = per muoversi

LISTATE CON NOI

PAINTER

Vuoi sbizzarrirti nel disegno? Eccoti accontentato. Digita il listato di questo mese e... vai col PAINTER!
Per disegnare premere il tasto '1', muovere il cursore col joystick premendo il tasto 'fire' per scrivere sul video.
Per tracciare una linea premere fire sul primo estremo, portare il cursore sul secondo e premere nuovamente fire. Per un rettangolo definire i due vertici opposti; per un cerchio dare il centro e un punto della circonferenza.
Per portare il cursore al centro dello schermo premere '5'; l'opzione '6' serve a colorare l'area attorno al cursore.
Scegliere uno dei 3 colori premendo il '7'. Per cambiare uno dei 3 colori premere il tasto corrispondente, e '+' o '-' per variarne la luminosità.
Per cancellare procedere come per creare un rettangolo, delimitando i vertici opposti dell'area.
Per cancellare tutto lo schermo premere Shift e Clear/Home.
Premendo 'c' si ha a disposizione tutto lo schermo.
N.B.: *Le lettere sottolineate vanno inserite come caratteri grafici:*
s = Shift-Clear/home
p = Control 1
Q = Cursore in giù

```
100 DIMX(8),Y(8),C(3),B(3):S=1:G=4
110 COLOR0,2,7:COLOR1,1,0:COLOR2,3,4:COLOR3,7,5:COLOR4,2,7:GRAPHIC4,1:GOTO210
120 GOSUB500:IFS=NTHEN190:ELSERETURN
130 V=0
140 FORI=0TOE:POKEI+P,V:NEXT:RETURN
150 GOSUB160:IFA$=" "THEN130:ELSEIFATHENRETURN:ELSE150
160 J=JOY(1):A=JAND128:GETA$:POKE239,0:I=(JAND15):Y=Y(I):X=X(I):Q=Q+X:D=D+Y
170 LOCATEQ,D:C=RDOT(2):DRAW(3-C),Q,D:DRAWC,Q,D:RETURN
180 FORT=1TO3:K=T:GOSUB200:NEXT:RETURN
190 K=0:T=S
200 FORI=0TO8:POKEI+2902+40*K,C(T)+B(T)*16-1:NEXT:RETURN
210 SCALE0:PRINT"spQQQQQQQQQQQQQQQQQQQQ1) DISEGNA  6) RIEMPI    B) CANCELLA
220 PRINT"2) TRACCIA  7) SORG. "S" C) SCHERMO":PRINT"3) RETT.    8) COLORE 1
230 PRINT"4) CERCHIO  9) COLORE 2":PRINT"5) CENTRO   A) COLORE 3";
240 IFZ=20THENGOSUB180:GOSUB190:RETURN
250 FORI=1TO3:READC(I),B(I):NEXT:Z=20:GOSUB180:N=1
260 Q=77:D=84:C(.)=1:C$="134s672B89AC5":FORI=0TO8:READX(I),Y(I):NEXT
270 GETKEYA$:POKE239,0:H=INSTR(C$,A$)
280 ONHGOSUB300,320,360,400,410,420,450,490,560,560,560,470,480
290 GOTO270
300 E=9:P=2848:V=66:GOSUB140
310 GOSUB150:IFVTHENDRAWS,Q,D:GOTO310:ELSERETURN
320 E=9:P=2928:F=0
330 V=66:GOSUB140:GOSUB150
340 IFV=0THENRETURN:ELSEV=66+128:GOSUB140:K=Q:R=D:GOSUB150:IFV=0THENRETURN
350 BOXS,K,R,Q,D,,F:IFFTHENRETURN:ELSE320
360 E=9:P=2968:V=66:GOSUB140:GOSUB150
370 IFV=0THENRETURN:ELSEV=66+128:GOSUB140:K=Q:R=D:GOSUB150:IFV=0THENRETURN
380 T=(K-Q)ç2:V=(R-D)ç2:U=1:W=1.6:IFT¨VTHENU=.625:W=1
390 T=INT(SQR(T+V)):CIRCLES,K,R,U*T,W*T:GOTO360
400 SCNCLR:GOSUB210:RETURN
410 E=8:P=2860:V=66:GOSUB140:GOSUB150:IFVTHENPAINTS,Q,D,1:GOTO410:ELSERETURN
420 E=10:P=2900:V=C(S)+B(S)*16-1:GOSUB140
430 GETKEYA$:IFA$="7"THEN440:ELSEIFA$=" "THENE=2:GOSUB130:RETURN:ELSE430
440 S=S+1+(S¨2)*3:POKE3934,S+48:GOTO420
450 E=9:P=2888:V=66:GOSUB140:GOSUB150:IFV=0THENRETURN
460 V=66:GOSUB140:K=Q:R=D:GOSUB150:IFVTHENDRAWS,K,RTOQ,D:GOTO450:ELSERETURN
470 IFG=4THENGRAPHIC3:G=3:RETURN:ELSEGRAPHIC4:G=4:GOSUB210:RETURN
480 Q=77:D=84:RETURN
490 F=1:E=13:P=2872:T=S:S=0:GOSUB330:S=T:IFV=0THENRETURN:ELSE490
500 FORI=0TO10:POKEI+O,C(N)+B(N)*16-1:NEXT:GETKEYA$:IFA$=" "THEN550
510 IFA$="-"THENM=1:GOTO530:ELSEIFA$="+"THENM=-1:GOTO530:ELSEIFA$¨¨N$THEN500
520 C(N)=(C(N)AND15)+1:GOTO540
530 B(N)=B(N)+MAND7
540 COLORN,C(N),B(N):GOTO500
550 FORI=0TO2:POKEI+O,0:NEXT:RETURN
560 N=H-8:N$=MID$("89A",N,1):O=2900+N*40:GOTO120
570 DATA1,,3,4,7,5,,,,-1,1,-1,1,,1,1,,1,-1,1,-1,,-1,-1
```

LISTATE CON NOI

bioritmi

MSX

Sei giù di corda e vuoi sapere quando finirà questo momentaccio? Ti senti in splendida forma fisica e vuoi sapere se è giunta l'ora per affrontare la scalata del monte Everest? Sei in un "periodo di grazia" e vuoi sapere per quanto tempo durerà?
Segui il tuo bioritmo e non potrai sbagliare.
Digita normalmente il listato poi, per leggere il responso, inserisci la tua data di nascita (giorno/mese/anno) e poi il mese e l'anno per il quale vuoi ottenere il responso.
Sullo schermo verrà visualizzato un grafico con 3 curve: quella dell'energia fisica (verde), quella del tuo stato emozionale (blu) e quella intellettuale (rossa).
Leggile, interpretale e... non sbaglierai più un colpo in vita tua!

```
1 REM COPYRIGHT 1985 BY M. CELLINI Q
10 SCREEN0:WIDTH36:KEYOFF:COLOR15,4,4
15 DIM P(31),E(31),I(31):PI=3.1415927#
20 LOCATE12,1:PRINT"- BIORITMI -"
30 LOCATE0,4:PRINT"INSERISCI LA DATA DI
NASCITA"
35 LOCATE0,7:INPUT"GIORNO:";J
40 LOCATE0,10:INPUT"MESE:";M
45 LOCATE0,13:INPUT"ANNO:";A
50 W=365*A+J+31*(M-1)+INT((A-1)/4)-INT(3
*(INT((A-1)/100)+1)/4)
60 Z=365*A+J+31*(M-1)-INT(.4*M+2.3)+INT(
A/4)-INT(3*(INT(A/100)+1)/4)
65 IF M<=2 THEN X=W
70 IF M>2 THEN X=Z
75 '*******************************
80 CLS:LOCATE12,1:PRINT"- BIORITMI -"
90 LOCATE0,4:PRINT"INSERISCI LA DATA DI
BIORITMO"
95 LOCATE0,7:INPUT"MESE:";M
100 LOCATE0,10:INPUT"ANNO:";A
105 J=1
110 W=365*A+J+31*(M-1)+INT((A-1)/4)-INT(
3*(INT((A-1)/100)+1)/4)
120 Z=365*A+J+31*(M-1)-INT(.4*M+2.3)+INT
(A/4)-INT(3*(INT(A/100)+1)/4)
125 IF M=2 AND A/4=INT(A/4) THEN N=29 EL
SE N=28
130 IF M=1 OR M=3 OR M=5 OR M=7 OR M=8 O
R M=10 OR M=12 THEN N=31
140 IF M=4 OR M=6 OR M=9 OR M=11 THEN N=
30
150 IF M<=2 THEN Y=W
155 IF M>2 THEN Y=Z
160 F=Y-X
200 '*******************************
210 SCREEN2
220 LINE(7,0)-(7,255)
230 LINE(7,96)-(255,96)
235 LINE(2,166)-(7,166):LINE(2,26)-(7,26
)
240 OPEN "GRP:" FOR OUTPUT AS#1
250 PRESET(25,2)
255 PRINT#1,"BIORITMI DEL MESE ";M;"/";A
260 PRESET(8,171)
265 PRINT#1,"12345678911111111122222222
2233"
270 PRESET(80,180)
275 PRINT#1,"0123456789012345678901"
280 CLOSE 1
300 FOR K=0 TO N-1
310 P(K)=70*SIN(2*PI*(K+F)/23)
320 E(K)=70*SIN(2*PI*(K+F)/28)
330 I(K)=70*SIN(2*PI*(K+F)/33)
340 NEXT K
350 FOR K=1 TO N-1
360 LINE(K*8,96+P(K-1))-((K+1)*8,96+P(K)
),3
370 LINE(K*8,96+E(K-1))-((K+1)*8,96+E(K)
),11
380 LINE(K*8,96+I(K-1))-((K+1)*8,96+I(K)
),9
390 NEXT K
400 IF STRIG(0) THEN RUN ELSE 400
```

IMPARIAMO L'ASSEMBLER

Undicesima lezione

di Onorio Ribolzi

Credo che, a questo punto del corso, diventa necessario chiarire formalmente una serie di concetti che, fino ad ora, sono stati accennati di sfuggita nel corso delle ultime lezioni.

ASSEMBLER E LINGUAGGIO MACCHINA

Salvo una puntualizzazione iniziale nella settima lezione, abbiamo utilizzato di fatto queste locuzioni come quasi sinonimi: poco male, fino ad ora, perché ci siamo limitati ad esaminare singole istruzioni e a presentare brevi esempi esplicativi.
Nell'uso pratico, invece, la differenza è abissale.
Per linguaggio macchina si intende, come abbiamo già visto diverse volte, una fila di numeri binari (che per nostra comodità si preferisce solitamente rappresentare in formato esadecimale) che il microprocessore per cui è stato scritto il programma è in grado di interpretare come istruzioni (e in questo caso i numeri si chiamano anche codici operativi) o dati su cui lavorare.
Affinché un programma in linguaggio macchina possa essere eseguito dal microprocessore andrà, ovviamente, caricato in memoria e successivamente bisognerà ordinare al microprocessore di iniziarne l'esecuzione a partire da una certa locazione di cui dovremo comunicare l'indirizzo.

In particolare, in ambiente BASIC, utilizzeremo una istruzione (SYS INDIRIZZO in Basic Commodore) che esegue il salto all'indirizzo iniziale del nostro programma attraverso una CHIAMATA A SUB-ROUTINE.
È per questo che le routines in linguaggio macchina destinate ad essere eseguite sotto il controllo di un programma Basic devono necessariamente terminare (salvo astuzie e 'contorcimenti' vari) con una istruzione di RETURN. Si narra che i programmatori dei primi computers lavorassero effettivamente in linguaggio macchina: in pratica, muniti di apposito corposo manuale, si scrivevano il programma un numerino dopo l'altro.
Ciò fatto, mettevano mano a un pacchetto di schede di cartoncino su cui trasferivano, sotto forma di perforazioni abilmente posizionate, tutta la collezione di codici operativi; in ultimo, le schede venivano infilate in un apposito aggeggio che provvedeva a leggersi i cartoncini 'ricamati' e a trasferirne i codici nella memoria dell'elaboratore.
Orrendo sistema, che illustra adeguatamente uno dei motivi per cui lavorare in linguaggio macchina, nudo e crudo, sia altamente sconsigliabile.

I PROGRAMMI MONITOR

Per un programmatore, la soluzione del problema non poteva non essere ovvia: istruire l'elaboratore stesso ad eseguire un compito noioso, meticoloso, ripetitivo.
Nascono in questo modo i programmi di traduzione, che si occupano di funzionare da interfaccia tra macchina e operatore.
Ad ogni codice operativo riconosciuto dall'elaboratore si associa una ISTRUZIONE MNEMONICA: ad esempio, il 6502 interpreta il codice AA come istruzione a Trasferire il dato presente in Accumulatore nel registro X: da cui il mnemonico TAX.
Dopo aver avuto questa geniale idea, il programmatore può anche dimenticarsi del significato dei vari codici operativi: gli sarà sufficiente valersi di un programma che funzioni per lui da interprete.
La categoria di programmi di questo genere più semplice e più diffusa negli home-computers viene detta MONITOR (da non confondersi con il monitor inteso come apparecchio televisivo specializzato dedicato ad essere usato con il computer).
Un buon Monitor potrebbe essere considerato una specie di interprete simultaneo. Di solito, sono disponibili come minimo, tra gli altri, i comandi Assembla, Disassembla, Memoria: i comandi riconosciuti dal Monitor sono solitamente costituiti da una singola lettera seguita eventualmente da alcuni parametri: i tre che abbiamo visto li chiameremo quindi A, D, M.
Tra questi il più semplice è il comando M, che mostra i valori depositati in una determinata area di memoria interpretato come una serie di numeri esadecimali.
Il comando D è più complesso: con questo infatti, è possibile disassemblare, ossia interpretare i dati presenti in un'area di memoria considerandoli codici operativi, e leggere sullo schermo i corrispondenti mnemonici.
Il comando A è complementare al precedente: permette di scrivere un breve programma utilizzando i mnemonici e ritrovarsi depositati in memoria i corrispondenti codici operativi.
La Commodore ha pensato bene di fornire finalmente le nuove macchine, a partire dalla serie C-16 / Plus 4 di un Monitor residente su ROM.
Si tratta di un programma di qualità non eccellente, che mette a disposizione giusto l'indispensabile, ma la comodità di poterne disporre senza dover caricare in memoria fa perdonare volentieri qualche carenza.
Per entrare in Monitor è sufficiente dare, appunto, il comando 'MONITOR' seguito da >RETURN<.
Per uscirne Tornando al Basic è sufficiente dare il comando 'X'.
Nel riquadro 1 troverete un esempio realizzato utilizzando appunto questo Monitor.

RIQUADRO 1

```
MONITOR
        PC    SR   AC   XR   YR   SP
   ;    FF00  00   00   FF   00   F9

M 9E12 9E20
-----------
>9E12 20  EB  B7  8A  A0  00  91  14  :R...
>9E1A 60  20  EB  B7  86  49  A2  00  :.I.$

D 9E12 9E1A
-----------

 . 9E12  20  D2  9D   JSR  $9DD2
 . 9E15  8A           TXA
 . 9E16  A0  00       LDY  #$00
 . 9E18  91  14       STA  ($14),Y
 . 9E1A  60           RTS
.
```

I comandi scritti da tastiera sono stati sottolineati: le tre prime righe rappresentano il prompt del Monitor e mostrano lo stato di tutti i registri del microprocessore nel momento in cui si è entrati in Monitor.
Per ora mi interessa soprattutto attirare la vostra attenzione sul fatto che quella che viene mostrata è la stessa zona di memoria, intepretata in modi differenti.
Non sempre, infatti, è ovvio che una sequenza di numeri presenti in memoria vada letta come parte di un programma: potrebbe trattarsi di un messaggio, di una immagine grafica, di una tabellina dei logaritmi o che altro volete.
Se state tentando di disassemblare un programma non vostro, facilmente incontrerete, inframezzati ai mnemonici, serie di punti interrogativi: significa che il dato in esame non corrisponde a nessuno dei codici operativi possibili; indice quasi sicuro del fatto che la zona di memoria che state decifrando non va interpretata come programma.
Una piccola nota in margine: la routine che vi abbiamo disassemblato è una parte del Basic Commodore, e viene utilizzata per eseguire l'istruzione POKE. Quando l'interprete Basic arriva in questo punto, ha già letto e riconosciuto la POKE, e si accinge a leggere i due parametri che la seguono: indirizzo e valore.
Quando avremo detto che la routine in 9DD2 ha appunto il compito di leggere un indirizzo, tradurlo nel formato standard LSB - MSB, depositarlo nelle locazioni di memoria $14-$15 e infine di leggere un dato compreso tra 0 e 255 e depositarlo nel registro .X, le poche istruzioni successive dovrebbero oramai esservi chiare.

MA NON È TUTTO!

Saper usare un Monitor è già un grosso passo avanti; finalmente potremo provare qualche piccolo programma, tentare di leggere i programmi altrui, farci un po' di esperienza.
Ma provatevi, dopo avere scritto una routine piuttosto lunga, a tentare di correggerla o migliorarla inserendo nel bel mezzo tre o quattro istruzioni in più; trattasi di un lavoro deprimente.
Bisognerà far slittare in giù di qualche byte tutta la seconda parte del programma, ricontrollare conseguentemente tutti i salti, inserire finalmente le proprie modifiche e poi rileggersi il tutto un paio di volte per assicurarsi di non aver commesso errori.
Il problema sta nel fatto che un Monitor non è in fondo che un modo più comodo per lavorare sui codici operativi depositati in memoria: nonostante tutto, si trascina dietro tutta la mancanza di flessibilità tipica del linguaggio macchina.
A noi invece piacerebbe poter scrivere in linguaggio macchina come si scrive un programma Basic: potendo inserire e togliere gruppi di istruzioni, magari costruendo col tempo una libreria di routines da utilizzare nei nostri prossimi programmi, mettendo là dove serve (in assembler praticamente ad ogni istruzione) qualche commento illuminante ecc. ecc.
Detto fatto. Quando inizia ad emergere una esigenza di questo genere (non appena si smette di pasticciare in via sperimentale) l'unica è passare ad un assemblatore.

GLI ASSEMBLATORI

L'idea di base su cui è costruito un assemblatore è molto semplice: separare in maniera radicale il testo del programma così come viene scritto dall'autore utilizzando mnemonici, commenti e tutte le altre prestazioni che vedremo, dalla sua traduzione in codici operativi, detta OGGETTO.
Normalmente un assemblatore è costituito da diversi moduli di programma progettati per svolgere funzioni completamente diverse: a volte questa filosofia è spinta al limite di fornire una serie di programmi separati, e caricabili nel computer uno per volta, dedicati ciascuno ad eseguire una parte del lavoro di assemblaggio.

La cosa risulta in pratica parecchio macchinosa, anche se ha il vantaggio di permettere maggiori prestazioni globali, ma noi faremo riferimento nel resto del discorso ad un assemblatore di questo tipo, che permette di fissare meglio le idee. Il primo modulo da caricare per iniziare a scrivere un programma è l'EDITOR: con questo potremo scrivere un testo utilizzando un formato predefinito leggibile dal programma successivo. Solitamente l'editor dispone di facilitazioni come AUTO-NUMBER, RENUMBER, cancellazione di blocchi di linee, ricerca e modifica di determinate parole, ecc. destinate propriamente a facilitare il lavoro di scrittura del testo. Attenzione! Non stiamo lavorando con un super monitor: per un programma editor l'istruzione TAX non è altro che una stringa di caratteri formata da SPAZIO,T,A,X,SPAZIO.

È responsabilità del programmatore seguire, nella stesura del testo, certe regole formali che permetteranno poi al programma successivo di leggere il sorgente e tradurlo in oggetto.
Questo programma successivo è, appunto, l'ASSEMBLATORE vero e proprio: dopo aver salvato il testo si carica e si lancia l'assemblatore. Se non avete commesso errori di sintassi, tutto bene: diversamente, dovrete annotarvi gli errori commessi, tornare in editor, correggere il testo sorgente e fare un'altra passata con l'assemblatore.

Alla fine di questo processo otterrete un OGGETTO: questo è un programma in linguaggio macchina caricabile ed eseguibile.
Non sembra molto attraente come faccenda, ma questa filosofia permette facilitazioni notevoli, tra cui due fondamentali: si tratta della possibilità di utilizzare ETICHETTE in sostituzione di indirizzi e valori e della possibilità di RILOCARE il programma, ossia di istruire l'assemblatore a scriverne una nuova versione in grado di girare in una diversa area di memoria.
Nel riquadro 2 trovate un brevissimo frammento di sorgente scritto con l'ASSEMBLER fornito dalla Commodore per il C16.

RIQUADRO 2

```
1000 ;QUESTA È UNA POSSIBILE
1010 ;VERSIONE DELLA ROUTINE DI
1030 ;ESECUZIONE DELL'ISTRUZIONE
1040 ;POKE CHE ABBIAMO VISTO IN
1050 ;PRECEDENZA
1060 ;
1070 GETNUM=$9DD2 ;PRENDE IND,NUM
1080 LINNUM=$14       ;PUNTATORE
1090 ZERO    =0
1100 ;
1110 * =$9E12
1120 ;
1130 POKE        JSR GETNUM
1140             TXA
1150             LDY #ZERO
1160             STA (LINNUM),Y
1170             RTS
```

Il simbolo ';' è l'equivalente della istruzione Basic REM, e permette di inserire un commento.
Inoltre, abbiamo utilizzato quattro etichette: a tre di queste abbiamo assegnato un valore esplicito utilizzando il comando (rivolto all'assemblatore) '=' la quarta, POKE, è solo scritta davanti all'istruzione. Quando l'assemblatore scorre il testo sorgente, memorizza tutte le etichette poste ad inizio riga attribuendogli come valore l'indirizzo assegnato alla istruzione letta subito dopo: in questo modo potremo, in un'altra parte del programma fare riferimento a POKE piuttosto che a un numero di linea o, peggio, ad un indirizzo.

Vediamo inoltre, subito prima del testo del programma vero e proprio, il comando (sempre rivolto all'assemblatore) "=': questo informa l'assemblatore che il programma andrà assemblato a partire dalla locazione di memoria di indirizzo $9E12.*
Deduzione: un intero programma in ASSEMBLER, se scritto utilizzando appropriatamente le etichette, può essere assemblato dovunque nella memoria semplicemente cambiando l'indirizzo iniziale. Rimuovendo il "=' può essere accodato a un programma principale che conterrà, ad esempio, una chiamata tipo*
JSR POKE
senza nessuna (o quasi) preoccupazione per gli indirizzi effettivi in cui andrà a collocarsi l'istruzione.
Se abbiamo utilizzato, come va fatto, etichette per definire la destinazione di tutti i salti relativi e assoluti potremo, in fase di miglioramento e correzione, apportare modifiche abbastanza tranquillamente.
Infine, un programma sorgente ben commentato resta leggibile anche dopo che ci siamo dimenticati i particolari di quel progetto...
Per questo mese ho abbondantemente esaurito lo spazio: i risultati degli esercizi al mese prossimo!